**Find computers Windows license status, 1 means active license**

Get-ADComputer -Filter \* | % {$\_.Name; Get-CimInstance -ComputerName $\_.Name -ClassName SoftwareLicensingProduct | where PartialProductKey | select Name, LicenseStatus}

**Clear an environment variable:**

clear-Item -Path Env:OS\_TENANT\_ID

Set an environment variable:

Set-Item -Path Env:$var\_name -Value $var\_value"

**Get an environment variable:**

get-item -path Env:OS\_PASSWORD

Display an enviroment variable

$Env:OS\_PASSWORD

**Create New Group**

New-ADGroup -Name "$ComputerName Admins" -SamAccountName "$ComputerName Admins" -GroupCategory Security -GroupScope Global -DisplayName "$ComputerName Admins" -Path "OU=Security Groups,$OU,OU=ADS,DC=ssfhs,DC=org" -Description "$ComputerName Admins"

**Create New User**

New-ADUser -SamAccountName hamid.maddi -Name "Hamid Maddi" -UserPrincipalName [hamid.maddi@oncaas.com](mailto:hamid.maddi@oncaas.com) -AccountPassword (ConvertTo-SecureString -AsPlainText "Interactive2014" -Force) -Enabled $true -PasswordNeverExpires $true -Path 'CN=Users,DC=oncaas,DC=com

**Add User to Group**

Add-ADGroupMember -Identity APPDEV -Member hamid.maddi

**ADD AD group to local AD Admin group**

$server = 'APPDEV-CIC-01'

$group = 'Administrators'

$user = 'oncaas/APPDEV'

$action = 'Add'

$objGroup = [ADSI]("WinNT://$server/$group")

$objADUser = [ADSI]("WinNT://$user")

$objGroup.PSBase.Invoke($action,$objADUser.PSBase.Path)

**Powershell profile setup on shared server:**

use $PsHome\Profile.ps1

**Powershell AD commands to fix chef issues:**

$filter="mgmt-wcmad\*"

Get-ADComputer -Filter "Name -like '$filter'" | ForEach-Object { $test=$\_.Name; dir [\\$test\c$\chef\client.rb](file:///\\$test\c$\chef\client.rb) }

Get-ADComputer -Filter "Name -like '$filter'" | ForEach-Object { $test=$\_.Name; rm [\\$test\c$\chef\client.rb](file:///\\$test\c$\chef\client.rb) }

Get-ADComputer -Filter "Name -like '$filter'" | ForEach-Object { $test=$\_.Name; copy-item .\client.rb -Destination [\\$test\c$\chef\](file:///\\$test\c$\chef\) }

Get-ADComputer -Filter "Name -like '$filter'" | ForEach-Object { invoke-command -ComputerName $\_.Name -ScriptBlock {chef-client}}

Get-ADComputer -Filter "Name -like '$filter'" | ForEach-Object { invoke-command -ComputerName $\_.Name -ScriptBlock {get-service chef-client}}